Pergamon

Computers Math. Applic. Vol. 31, No. 4/5, pp. 173–178, 1996

0898-1221/96 $15.00 + 0.00

0898-1221(95)00229-4

The Convergence Speed of Interval Methods for Global Optimization

A. E. CSALLNER AND T. CSENDES
József Attila University, Institute of Informatics
Szeged, Hungary
<csallner><csendes>@inf.u-szeged.hu

Abstract—Three particular algorithms from a class of interval subdivision methods for global optimization are studied. The theoretical upper bound on the convergence speed of Hansen's method is given. The three methods (by Hansen, Moore-Skelboe, and a new one with a random actual box selection rule) are compared numerically.

Keywords—Global optimization, Interval arithmetic, Subdivision.

1. PROBLEM DEFINITION

The theory of applying interval arithmetic for global optimization problems has a history of about two decades, but even in spite of the capabilities of the new supercomputers they are not extensively used for real-life problems because they are sometimes very slow (with exponentially bounded convergence speed). These methods are applicable for almost all NLP problems to solve them reliably, and this property makes them still so interesting.

In this paper, we consider exclusively unconstrained global optimization problems of the form

$$\min_{x \in X} f(x),$$

where $f : \mathbb{R}^n \to \mathbb{R}$ is the objective function, $X \in \mathbb{I}^n$ is an n-dimensional interval (also called box), and $\mathbb{I}$ stands for the set of compact real intervals. No special problem structure is required: only the inclusion function of the objective function is utilised [1]. We call a function $F : \mathbb{I}^n \to \mathbb{I}$ to be an *inclusion function* of f, if $x \in Y$ implies $f(x) \in F(Y)$ for each interval Y in X. In other words, $\hat{f}(Y) \subseteq F(Y)$, where $\hat{f}(Y)$ is the range of f on Y. In the following, we denote the width of an interval Y by $w(Y)$, the lower bound of a one-dimensional interval Z by $\operatorname{lb} Z$, and the upper bound by $\operatorname{ub} Z$. It is assumed that for all the inclusion functions,

$$w\left(F\left(X^i\right)\right) \to 0 \qquad \text{as } w\left(X^i\right) \to 0 \tag{1}$$

holds.

Most of the basic interval subdivision methods [2–8] for unconstrained global optimization have the same outline. In the following, we show one of the possible definitions of the bisection methods for unconstrained global optimization.

ALGORITHM 1.1. (General interval bisection algorithm for global optimization)

1. Set $Y := X$.
2. Initialize list $L = \{Y\}$.

This work was supported by the Hungarian Academy of Sciences and by the Grants OTKA 2879/1991 and MKM 414/1994.

Typeset by $\mathcal{A}\mathcal{M}\mathcal{S}$-TEX

3. Choose a coordinate direction for the bisection.
4. Bisect Y normal to the chosen coordinate direction in a given ratio getting the subintervals Y_1 and Y_2.
5. Insert Y_1 and Y_2 into the list under certain conditions.
6. Denote one of the elements on the list by Y and discard it from the list.
7. Terminate under certain conditions.
8. Go to 3.

To have a well-defined method, we have to fix the meaning of the individual steps of Algorithm 1.1 which above are only circumscribed. In Step 3, a coordinate direction normal to which the bisection is made has to be chosen. The bisection ratio, which is usually half to half, is to be fixed in Step 4. In Step 5, the conditions have to be made clear under which the reinsertion of the subintervals Y_1 and Y_2 into the list makes sense. An interval selection rule decides in Step 6 which of the elements of the list is to be bisected next. Finally, the termination criterion shows whether we can stop or have to start a new iteration.

The three methods we compare in this paper differ mainly in the selection rule they apply in Step 6 of Algorithm 1.1. Although depending on other above listed modifications, several versions of these methods use the same principle in choosing an interval from the list for the next iteration step. The method of Moore and Skelboe [1,6,9] selects the box with the smallest inclusion function lower bound value and the method due to Hansen [1,3] the box with the greatest width. In the next section, we determine the convergence speed of the latter method in general.

2. CONVERGENCE SPEED OF HANSEN'S ALGORITHM

It has already been proved that the Moore-Skelboe algorithm's convergence speed is exponential when assuming isotonicity for the inclusion functions applied for the algorithm [7], where the *isotonicity* property means that $Y \subseteq Z$ implies $F(Y) \subseteq F(Z)$ (see [1]). It has also been shown on the same place that without isotonicity the convergence speed can be arbitrarily slow.

DEFINITION 1. *An inclusion function F of f over X is called of (convergence) order α if $w(F(Y)) - w(\hat{f}(Y)) \leq cw(Y)^\alpha$ holds for all intervals in X and some constants c, $\alpha > 0$.*

The next theorem says that using the interval selection rule of Hansen instead of that from the Moore-Skelboe algorithm, exponential convergence can be achieved without assuming isotonicity.

THEOREM 1. *If F is an inclusion function of order α of f, then*

$$\operatorname{lb} \hat{f}(X) - \operatorname{lb} F(Y_k) \leq c \cdot (2w(X))^\alpha \cdot (k+1)^{-\alpha/n}, \tag{2}$$

where Y_k stands for the box to be bisected in the k^{th} iteration step.

To prove this theorem, we define the level concept through which the proof becomes simple.

DEFINITION 2. *Subdividing the boxes uniformly, we define an iteration step to be on the l^{th} level if the maximum size of the boxes on the list is $2^{-l}w(X)$.*

REMARK 1. If k and $k+1$ are iteration steps and k is on the l^{th} level, then $k+1$ is either on level l, or on level $l+1$.

After this definition, three trivial lemmas follow immediately to introduce the proof of Theorem 1.

LEMMA 1. *Completing the iteration steps of level l, the number of boxes N_l on the list becomes 2^n-fold, $N_{l+1} = 2^n N_l$.*

LEMMA 2. *The number of the iteration steps involved in the l^{th} level equals the initial number of subboxes on that level multiplied by $2^n - 1$, where from Lemma 1 the number of boxes equals 2^{ln}. Thus, $2^{ln}(2^n - 1)$ iteration steps belong to the l^{th} level.*

LEMMA 3. *The following two assertions hold for the iteration steps on the l^{th} level:*

(i) $2^{ln} - 1 \le k \le 2^{(l+1)n} - 1$,

(ii) $(k+1)^{-1/n} \le 2^{-l} < 2(k+1)^{-1/n}$.

To prove our theorem, we need only one more assertion which is based simply on the inclusion property and the α-convergence of the inclusion function F.

PROPOSITION 1. *The difference of the global minimum and the lower bound of the inclusion function value $F(Y_k)$ is bounded from above as*

$$\mathrm{lb}\, \hat{f}(X) - \mathrm{lb}\, F(Y_k) \le cw^{\alpha}(Y_k).$$

PROOF. The assertion follows immediately from the simple calculation

$$\begin{aligned}\mathrm{lb}\, \hat{f}(X) - \mathrm{lb}\, F(Y_k) &\le \mathrm{lb}\, \hat{f}(Y_k) - \mathrm{lb}\, F(Y_k) = \mathrm{ub}\, \hat{f}(Y_k) - w\left(\hat{f}(Y_k)\right) - \mathrm{ub}\, F(Y_k) + w(F(Y_k)) \\ &\le w(F(Y_k)) - w\left(\hat{f}(Y_k)\right) \le cw^{\alpha}(Y_k).\end{aligned}$$

∎

PROOF OF THEOREM 1. Substituting the right-hand side of Proposition 1 first for the definition of the l^{th} level, and then for assertion (ii) of Lemma 3, we get the inequality

$$\mathrm{lb}\, \hat{f}(X) - \mathrm{lb}\, F(Y_k) \le cw^{\alpha}(X)2^{\alpha}(k+1)^{-\alpha/n},$$

and this is what we wanted to prove. ∎

3. NUMERICAL COMPARISON

Despite of the appealing theoretical results of the previous section, Hansen's algorithm realizes a branch and bound strategy with breadth first search. Thus, the number of subintervals to be subdivided at each level of iteration steps grows exponentially for any optimization problem. Hence, it is reasonable to compare the investigated procedures on a set of often used particular test problems.

For the tests, eleven test problems have been used, namely:

1. Levy No. 1 [10],
2. Levy No. 2 [10],
3. Rosenbrock function [11],
4. Three hump camel function [12],
5. Booth problem [11],
6. Matyas problem [11],
7. Powell problem [11],
8. Dripstone cave function ($n = 2$), where the dripstone cave function is of the form

$$f(x) = \left(\frac{1}{n}\sum_{i=1}^{n} \sin(x_i)\right)\left(1 - \frac{4}{441\pi^2 n}\sum_{i=1}^{n}\left(x_i - 19\frac{\pi}{2}\right)^2\right),$$

9. Dripstone cave function ($n = 3$),
10. Dripstone cave function ($n = 4$),
11. Dripstone cave function ($n = 5$).

We tested three interval selection rules:

1. the Moore-Skelboe rule (implemented with a partially ordered list),
2. Hansen's rule (with a first-in-first-out data structure), and
3. random selection.

All the three considered algorithms have contained the common specialities to Algorithm 1.1 (except for the interval selection rule). In Step 3, the coordinate with the longest edge has been chosen. We have always halved the box Y in Step 4. First, the midpoint test [1] was employed in Step 5 to investigate the behaviour of the interval bisection methods on initial boxes with a greater volume and with sharper termination criterion; subsequently, tests were made without the midpoint test. The termination criterion used in all cases was the following:

$$\text{Terminate if } \mathit{midtest} - \mathit{lobnd} < \epsilon, \tag{3}$$

where *midtest* denotes the function value at the middle of the actual interval, and *lobnd* stands for the smallest lower bound inclusion function value on the boxes of the list. The values used for ϵ are given in the tables below. All computer programs for the tests have been written in C++. The inclusion functions have been computed automatically with natural interval extension (all isotone and of order $\alpha = 1$).

The following three tables contain the number of function evaluations and the maximum number of list elements during the tests. For the version with the random interval selection rule, average and best figures of ten independent executions are given. The third column of each table contains the volume of the initial boxes.

Table 1. Number of function evaluations using the midpoint test.

Problem specification			Methods			
#	ϵ	Volume	1	2	3 (average)	3 (best)
1	5E−02	8	10617	11351	–	–
2	1E−04	20	2402	3016	–	–
3	1E−18	1E6	934	3918	3410	2956
4	1E−18	5	1370	2101	2130	1960
5	1E−18	1E14	1097	1926	2431	1999
6	1E−04	400	18089	19317	33559	27026
7	1E−07	1E4	1489	10614	12635	10354
8	1E−18	3969	231	941	1173	869
9	1E−18	250047	353	4573	4789	2850
10	1E−18	15752961	472	23988	21938	18070
11	1E−18	992436543	591	–	–	–

Table 2. Maximum number of list elements using the midpoint test.

Problem specification			Methods			
#	ϵ	Volume	1	2	3 (average)	3 (best)
1	5E−02	8	1830	1475	–	–
2	1E−04	20	413	357	–	–
3	1E−18	1E6	120	87	62	41
4	1E−18	5	91	37	26	22
5	1E−18	1E14	139	11	18	13
6	1E−04	400	826	483	589	454
7	1E−07	1E4	196	321	254	213
8	1E−18	3969	32	20	25	17
9	1E−18	250047	55	122	104	77
10	1E−18	15752961	75	671	684	490
11	1E−18	992436543	95	–	–	–

Table 3. Maximum number of list elements without the employment of the midpoint test.

Problem specification			Methods	
#	ϵ	Volume	1	2
1	5E−2	8	2929	–
2	1E−1	20	43	1495
3	1E−3	1	13	511
4	5E−2	5	99	721
5	5E−3	20	15	298
6	5E−2	4	115	1023
7	2	625	30	173
8	1E−4	25	8	255
9	1E−2	125	10	1023
10	5E−2	15752961	8	135
11	5E−2	992436543	10	527

The initial boxes have been chosen in some cases to avoid global optimizer points arising on the border of them. The parameter ϵ was determined simply to stop after an acceptable amount of iteration steps in most cases. Although the values of ϵ and the initial box volume are different from problem to problem, this fact does not hinder the fair comparison of the methods, because ϵ and $w(X)$ are kept constant for each individual problem.

In the tables, test results are missing (denoted by a '–') where the list overflowed during the computation or by the third method where the list overflowed in several cases and the difference between the results was so big that they would have not given much information. On that account, the last two columns of Table 3 are missing.

According to these preliminary tests, the first interval selection rule (by Moore and Skelboe) seems to be the most efficient. However, the usage of the midpoint test (Tables 1,2) could improve the second method to such a degree that it produced shorter lists during the execution than the first algorithm for the first six test problems. Moreover, the midpoint test can also decrease the number of function evaluations for Hansen's method substantially—while it does not influence that of the Moore-Skelboe algorithm. This fact is the explanation why the procedure of Hansen without the midpoint test (Table 3) provided incomparably worse performance in terms of number of function evaluations, as well (no additional table for the number of function evaluations is needed). All in all, the usage of the midpoint test decreased the difference between the efficiencies of the Moore-Skelboe and the Hansen methods (Tables 1–3).

Further investigations are in progress to find new interval selection rules exploiting the first and second derivatives of the objective function on one hand, and to understand the joint effects of several modifications on Algorithm 1.1 on the other hand.

4. CONCLUDING REMARKS

In this paper, we compared two often used and one new bisection procedure for global optimization. The second section showed that Hansen's algorithm can achieve the same order of convergence as the Moore-Skelboe method—without assuming the isotonicity of the inclusion function involved. The numerical tests in the third section made us conscious of the weak efficiency in the average case with the selection rule of the Hansen algorithm. The method using random selection proved to be no better than the other two procedures in general. Further investigations are required to clarify how far the suggested modifications on the general algorithm influence the efficiency of the bisection methods on wider classes of problems.

REFERENCES

1. H. Ratschek and J. Rokne, *New Computer Methods for Global Optimization*, Ellis Horwood, Chichester, (1988).
2. T. Csendes and J. Pintér, The impact of accelerating tools on the interval subdivision algorithm for global optimization, *European J. of Operational Research* **65**, 314–320 (1993).
3. E.R. Hansen, Global optimization using interval analysis—The multidimensional case, *Numerische Mathematik* **34**, 247–270 (1980).
4. E. Hansen, *Global Optimization Using Interval Analysis*, Marcel Dekker, New York, (1992).
5. C. Jansson and O. Knüppel, A global minimization method: The multi-dimensional case, Report 92.1, Technische Universität Hamburg-Harburg, (1992).
6. R.E. Moore, *Interval Analysis*, Prentice Hall, Engelwood Cliffs, NJ, (1966).
7. H. Ratschek and J. Rokne, Efficiency of a global optimization algorithm, *SIAM Journal on Numerical Analysis* **24**, 1191–1201 (1987).
8. D. Ratz, Automatische Ergebnisverifikation bei globalen Optimierungsproblemen, Dissertation, Universität Karlsruhe, (1992).
9. S. Skelboe, Computation of rational interval functions, *BIT* **14**, 87–95 (1974).
10. J.J. Moré, B.S. Garbow and K.E. Hillstrom, Testing unconstrained optimization software, *ACM Trans. Math. Software* **7**, 17–41 (1981).
11. A.V. Levy, A. Montalvo, S. Gomez and A. Calderon, Topics in global optimization, *Lecture Notes in Mathematics #909*, Springer, Berlin, (1981).
12. L.C.W. Dixon and G.P. Szegö, *Towards Global Opimization*, North-Holland, Amsterdam, (1975).